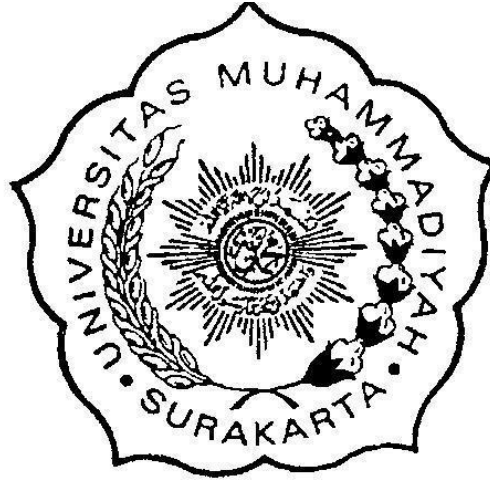


**APLIKASI PEMBELAJARAN JUZ AMMA
UNTUK ANAK USIA DINI BERBASIS ANDROID**



**Disusun sebagai salah satu syarat memperoleh Gelar Strata I
pada Jurusan Informatika Fakultas Komunikasi dan Informatika**

Oleh:

IVANOVITCZ ABDURRAKHMAN AR RANIRI
L200170153

**PROGRAM STUDI INFORMATIKA
FAKULTAS KOMUNIKASI DAN INFORMATIKA
UNIVERSITAS MUHAMMADIYAH SURAKARTA
2021**

HALAMAN PERSETUJUAN

**APLIKASI PEMBELAJARAN JUZ AMMA
UNTUK ANAK USIA DINI BERBASIS ANDROID**

PUBLIKASI ILMIAH

oleh:

IVANOVITCZ ABDURRAKHMAN AR RANIRI
L200170153

Telah diperiksa dan disetujui untuk diuji oleh:
Dosen Pembimbing



Dr. Endah Sudarmilah, S.T, M.Eng
NIK.969

HALAMAN PENGESAHAN

APLIKASI PEMBELAJARAN JUZ AMMA UNTUK ANAK USIA DINI BERBASIS ANDROID

OLEH
IVANOVITCZ ABDURRAKHMAN AR RANIRI
L200170153

Telah dipertahankan di depan Dewan Penguji
Fakultas Komunikasi dan Informatika
Universitas Muhammadiyah Surakarta
Pada hari Rabu, 30 Juni 2021
dan dinyatakan telah memenuhi syarat

Dewan Penguji :

1. Dr. Endah Sudarmilah, S.T, M.Eng
(Ketua Dewan Penguji)
2. Aris Rakhmadi, S.T., M.Eng
(Anggota I Dewan Penguji)
3. Heru Supriyono, S.T, M.Sc, PhD
(Anggota II Dewan Penguji)




(.....)

(.....)

(.....)

Dekan
Fakultas Komunikasi dan Informatika




Nurgiyatna, S.T, M.Sc, Ph.D
NIK. 881

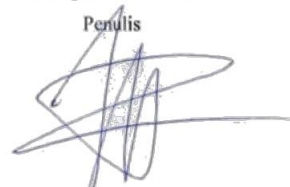
PERNYATAAN

Dengan ini saya menyatakan bahwa dalam publikasi ilmiah ini tidak terdapat karya yang pernah diajukan untuk memperoleh gelar kesarjanaan di suatu perguruan tinggi dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan orang lain, kecuali secara tertulis diacu dalam naskah dan disebutkan dalam daftar pustaka.

Apabila kelak terbukti ada ketidakbenaran dalam pernyataan saya di atas, maka akan saya pertanggungjawabkan sepenuhnya.

Salatiga, 25 Juni 2021

Penulis



IVANOVITCZ ABDURRAKHMAN AR RANIRI

L200170153

APLIKASI PEMBELAJARAN JUZ AMMA UNTUK ANAK USIA DINI BERBASIS ANDROID

Abstrak

Anak usia dini mudah bosan dan lelah dengan pembelajaran monoton dan hanya berisi tulisan yang menumpuk berbaris-baris, pembelajaran yang menyenangkan dan penuh warna akan lebih menarik perhatian anak dalam setiap kegiatan belajarnya. Hal ini perlu disiasati dengan pembelajaran yang lebih menarik, tidak hanya mengandalkan tulisan namun juga menampilkan ilustrasi yang menarik perhatian dan rasa keingintahuan anak. Begitu pula dengan mengajarkan anak-anak untuk mengaji, seperti pada Kamus Besar Bahasa Indonesia mengaji adalah mendaras (membaca) *Al-Quran*. Anak-anak perlu hal menarik agar mereka mau mengaji dengan senang hati tanpa adanya paksaan. Suasana yang menyenangkan tentu saja penting agar bisa menstimulasi anak belajar mengaji dengan semangat. Maka dari itu diperlukanlah media pembelajaran dengan ilustrasi yang menarik dengan tujuan anak-anak tertarik untuk mengaji dan tidak mudah bosan saat mengaji. Aplikasi ini menampilkan berbagai ilustrasi dari setiap Surah yang ada di *Al-Quran Juz 30* atau yang lebih sering disebut *Juz Amma*. Setiap *surah* akan terdapat ilustrasi yang menggambarkan arti *surah* tersebut. Beberapa fitur pada aplikasi ini adalah keterangan *surah*, arti tiap *ayat*, dan suara tiap *ayat*, data-data tersebut didapatkan dari API (*Application Programming Interface*) yang terhubung dengan data *Al-Quran* yang disediakan oleh Kementerian Agama Indonesia pada *website quran.kemenag.go.id*. Orang tua atau orang yang mengajari mengaji juga bisa memantau pembelajaran anak di aplikasi tersebut melalui *Bot Telegram* atau *website* yg terhubung dengan aplikasi dengan memasukkan *unique ID* yg didapatkan di aplikasi. Metode pengembangan perangkat lunak yang digunakan adalah model *waterfall*. Aplikasi ini dibuat menggunakan *framework* bahasa pemrograman Dart, yaitu Flutter dan ditargetkan untuk sistem operasi Android diatas versi 5.0 atau *lollipop*. Hasil pengembangan aplikasi ini diharapkan dapat membuat anak usia dini lebih tertarik untuk mengaji.

Kata Kunci: android , aplikasi, juz amma

Abstract

Early childhood is easily bored and tired of monotonous learning and only contains lines of text piling up, fun and colorful learning will attract children's attention more in every learning activity. This needs to be dealt with with more interesting learning, not only relying on text but also displaying illustrations that attract children's attention and curiosity. Likewise, teaching children to recite the Al-Quran, as in the Kamus Besar Bahasa Indonesia, reciting the Al-Quran is reciting (reading) the Al-Quran. Children need something interesting so that they will be happy to recite the Al-Quran without coercion. A pleasant atmosphere is of course important in order to stimulate children to learn the Al-Quran with enthusiasm. Therefore, a learning media with attractive illustrations is needed with the aim that children are interested in the Al-Quran and are not easily bored when

reciting the Al-Quran. This application displays various illustrations of each Surah in the Al-Quran Juz 30 or more commonly called Juz Amma. Each surah will have an illustration that illustrates the meaning of the surah. Some of the features in this application are the description of the surah, the meaning of each verse, and the sound of each verse, these data are obtained from the API (Application Programming Interface) which is connected to Al-Quran data provided by the Indonesian Ministry of Religion on the quran.kemenag.go.id website. Parents or people who teach the Al-Quran can also learn about children's learning in the application via the Telegram Bot or website linked to the application by entering the unique ID obtained in the application. The software development method used is the waterfall model. This application is created using the Dart programming language framework, namely Flutter and is targeted for the Android operating system above version 5.0 or lollipop. The results of the development of this application hope to make early childhood more interested in the Al-Quran.

Keywords: android , application, juz amma

1. PENDAHULUAN

1.1 Latar Belakang

Anak usia dini lebih senang bermain daripada harus mempelajari sesuatu. Proses belajar pada anak-anak harus dilakukan dengan kegiatan menyenangkan, karena dunia anak adalah dunia bermain. Oleh karena itu, diperlukannya metode belajar yang menyenangkan bagi anak-anak, salah satunya adalah dengan media pembelajaran interaktif.

Dewasa ini, pendidikan telah mencakup dunia digital media. Kegiatan pembelajaran telah bergerak ke arah penggunaan metode digital media interaktif dalam penyampaian materi (Sudarmilah & Negara, 2013). Media pembelajaran interaktif dapat membantu seorang anak dalam memahami pelajaran yang diberikan, metode pembelajaran untuk anak usia dini juga belajar sambil bermain. (Rahayu & Fujiati, 2018). Dibandingkan tulisan yang berbaris-baris, diperlukan adanya gambar ilustrasi yang menggambarkan makna dari tulisan tersebut. Seperti halnya dalam buku cerita, gambar ilustrasi menjadi bagian yang hampir tak terpisahkan dalam buku cerita untuk anak-anak. Gambar ilustrasi juga membantu anak mengimajinasikan bagaimana maksud dari tulisan. Kolaborasi antara tulisan dan gambar sama-sama pentingnya dalam penyampaian maksud cerita kepada anak. Ilustrasi gambar akan lebih mengeksplorasi imajinasi mereka dari tulisan yang mereka baca, sehingga tulisan tersebut terlihat lebih mudah memvisualisasikan apa yang dimaksud, seperti jika

mereka tidak paham dengan penjelasan tulisan tersebut, gambar ilustrasi akan membantu anak-anak untuk menjelaskan maksud dari tulisan tersebut.

Begitu pula dengan mengajarkan anak untuk mengaji, kadang kali anak malas atau bahkan tidak mau mengaji. Agar menyenangkan dan menarik bagi mereka, media untuk mengaji perlu juga ditambahkan ilustrasi yang menarik perhatian anak-anak. Ilustrasi seperti gambaran bagaimana makna dari tiap *surah*, ilustrasi ini tentu saja akan membantu anak untuk lebih memahami makna *surah* tersebut daripada hanya menggunakan tulisan saja.

Menurut uraian di atas, perlu dikembangkan aplikasi pembelajaran interaktif untuk menarik minat anak mengaji. Karena aplikasi ini ditujukan untuk anak usia dini, maka *surah-surah* yang ditampilkan hanya *surah* pada *juz 30* atau lebih sering disebut dengan *Juz Amma*. Selain sebagai media pembelajaran interaktif, aplikasi ini juga membantu orangtua atau orang yang mengajari mengaji untuk memantau kegiatan mengaji anak melalui *Bot* pada aplikasi Telegram atau melalui website yang terhubung dengan aplikasi dengan cara memasukkan *unique ID* yang didapatkan di dalam aplikasi.

Aplikasi ini dikembangkan untuk platform Android dengan versi minimal 5.0 atau *lollipop*. Android dipilih karena mayoritas masyarakat di Indonesia memilih menggunakan sistem operasi Android untuk *smartphone* mereka, seperti yang dilansir pada *website* <https://gs.statcounter.com> menyebutkan bahwa *Mobile Operating System Market Share* Indonesia selama bulan Januari 2020 sampai Maret 2021 sistem operasi Android menduduki peringkat pertama dengan nilai persentase sebesar 92,13% dan *Mobile Android Version Market Share* Indonesia selama bulan Januari 2020 sampai Maret 2021 sistem operasi Android dengan versi 5.0 dan 5.1 memiliki nilai persentase sebesar 8,04% dan 1,05%. Oleh karena itu, dengan mempertimbangkan masih adanya pengguna Android versi 5.0 dan 5.1, maka dipilihlah Android dengan versi minimal 5.0 sebagai media implementasi aplikasi pembelajaran *juz amma* ini.

1.2 Tinjauan Pustaka

1.2.1 Sistem Operasi Android

Android adalah sistem operasi *Open Source* didasarkan pada kernel Linux di mana aplikasi berjalan pada kerangka aplikasi yang mengontrol aktivitas dan didukung oleh *libraries* dan

Dalvik virtual machine yang mengompilasi serta mengubah semua *java class files* menjadi satu file (Haris et al., 2018). Sampai sekarang ini Android masih menjadi sistem operasi yang banyak diminati oleh masyarakat karena mudah digunakan dan juga *Open Source*, yang membuat sistem operasi ini bebas untuk dimodifikasi serta didistribusikan kembali oleh perusahaan lain. Android sendiri merupakan sistem operasi dari modifikasi kernel Linux. Selain dari kernel Linux sendiri, Android dilisensikan dengan lisensi yang ramah bisnis (Apache/MIT/BSD) sehingga siapapun dapat dengan bebas mengembangkannya dan menggunakannya untuk berbagai tujuan, bahkan untuk tujuan komersial. Android telah mengalami beberapa kenaikan versi sejak dirilis. Umumnya nama dari versi Android berasal dari nama makanan penutup, seperti *Cupcake* untuk Android versi 1.5 sampai *Pie* untuk Android versi 9.0.

1.2.2 Flutter

Flutter adalah *cross-platform framework* yang menargetkan pengembangan aplikasi *mobile* berkinerja tinggi. Flutter dirilis secara publik pada tahun 2016 oleh Google (Tashildar et al., 2020). *Framework* dari bahasa pemrograman Dart ini pada versi 2, yaitu Flutter 2 sudah mendukung pengembangan aplikasi yang *multi-platform*, bisa digunakan diberbagai platform seperti Android, IOS, *website*, dan bahkan sudah bisa digunakan untuk pengembangan aplikasi *desktop* untuk Windows, MacOS, dan Linux. Banyak perusahaan yang sudah mulai mengadopsi Flutter dalam pengembangan aplikasi mereka, dalam Flutter *Engage* yang diadakan pada 3 Maret 2021 disebutkan Microsoft menggunakan Flutter dalam pengembangan *smartphone foldable* mereka, yaitu Surface Duo.

1.2.3 Bot Telegram

Telegram adalah sebuah sistem perpesanan yang lintas platform dan berpusat pada keamanan kerahasiaan pribadi penggunanya, sedangkan bot adalah program komputer yang melakukan pekerjaan tertentu secara otomatis (Utomo et al., 2017). *Bot* pada Telegram merupakan aplikasi pihak ketiga yang berjalan di dalam Telegram. *Developer* dari *Bot* Telegram dapat mengontrol *Bot* menggunakan *HTTPS request* melalui *Bot API* yang terhubung ke server mereka. Banyak fitur dari *Bot* Telegram ini, seperti *developer* dapat mengirimkan notifikasi dan berita berkala secara otomatis, menerima pembayaran

menggunakan *Bot Payments* API, dan bahkan untuk mengembangkan permainan *multiplayer* sederhana.

1.2.4 Aplikasi Pembelajaran *Juz Amma* Berbasis Android

Penelitian terdahulu ini mengembangkan aplikasi pembelajaran *Juz Amma* untuk dioperasikan menggunakan sistem operasi Android, aplikasi yang digunakan dalam pengembangan adalah Adobe Flash CS6, Edtaw, dan Adobe Photoshop. Hasil aplikasi yang dikembangkan cukup sederhana sehingga memudahkan pengguna untuk mempelajari *surah-surah* di *juz amma*, namun pada bagian tampilan aplikasi terlihat tidak *responsive* dan beberapa warna yang digunakan sedikit mencolok serta kurang kontras (Firdaus, 2016).

1.2.5 Sistem Aplikasi Pembelajaran *Juz Amma* Menggunakan Web Mobile untuk Meningkatkan Meningkatkan Minat Hafalan Surat Pendek Siswa MI Miftahul Huda Sendang Agung

Penelitian terdahulu ini mengembangkan aplikasi menggunakan Construct 2 dan aplikasi pendukung seperti Corel Draw X7 dan Audacity. Hasil dari aplikasi yang dikembangkan mempermudah dalam belajar menghafal *juz amma*, selain fitur *juz amma* aplikasi tersebut juga terdapat fitur doa harian dan kisah para nabi. Tampilan dari aplikasi cukup sederhana dan terlihat *responsive*, namun perlu ditambahkan beberapa animasi agar lebih menarik perhatian pengguna, terutama untuk pengguna anak-anak (Andoyo et al., 2019).

1.2.6 Aplikasi Interaktif *Tahfidz Al-Quran Juz Amma*

Penelitian terdahulu ini mengembangkan Aplikasi Interaktif *Tahfidz Al-Quran Juz Amma*. Hasil dari aplikasi tersebut dapat membantu meningkatkan kualitas hafalan siswa dalam pembelajaran *tahfidz Al-Quran juz amma*. Tampilan dari aplikasi tersebut cukup menarik, aplikasi tidak hanya berupa teks saja, namun juga diberikan ilustrasi animasi yang menarik perhatian (Solihin, 2020).

2. METODE

Metode pengembangan perangkat lunak yang digunakan adalah *System Development Life Cycle* (SDLC). SDLC adalah metodologi untuk merancang, membangun, dan memelihara informasi dan sistem industri (Navita, 2017). Model SDLC yang dipilih dalam pengembangan perangkat lunak ini adalah model *waterfall*. Model *Waterfall* adalah model

proses pertama yang dibuat dan diperkenalkan oleh Winston W. Royce pada tahun 1970 (Iqbal & Idrees, 2017). *Waterfall* adalah salah satu model paling sederhana untuk dipahami dan digunakan. Model ini ditentukan oleh serangkaian langkah yang harus diselesaikan dalam urutan linier atau sekuensial (Agarwal et al., 2017). *Waterfall* menggambarkan proses pengembangan perangkat lunak secara aliran linier yg berurutan, maka dari itu model *waterfall* disebut sebagai *linear-sequential life cycle model*, artinya tahapan apapun dalam proses pembangunan dimulai hanya jika tahapan sebelumnya selesai. *Waterfall* dipilih dalam pengembangan aplikasi ini karena sederhana untuk dipahami dan digunakan, serta cocok untuk aplikasi skala kecil.

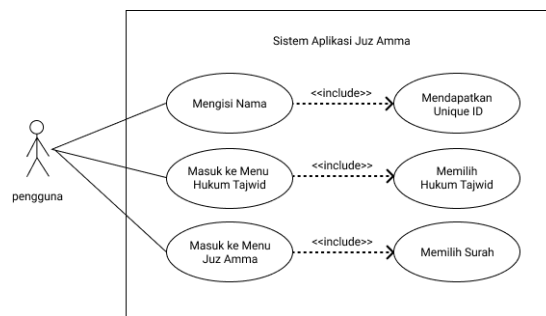
2.1 Requirements Analysis

Tahapan awal dari model *waterfall* adalah *Requirements Analysis*, pada tahapan ini mengumpulkan kebutuhan dan menganalisis semua kebutuhan apa saja yang akan diperlukan nantinya dalam pengembangan perangkat lunak, yang pada pengembangan ini adalah Aplikasi Pembelajaran *Juz Amma* untuk Anak Usia Dini Berbasis Android. Kebutuhan dalam pengembangan aplikasi ini adalah kebutuhan *hardware* dan kebutuhan *software*. Kebutuhan *hardware* dalam pengembangan aplikasi ini adalah sebuah laptop dengan spesifikasi prosesor Intel i3-6006U *dual-core* 2GHz, RAM 8 GB, dan HDD 1TB, serta sebuah *smartphone* Android yang dibutuhkan untuk menjalankan aplikasi selama proses pengembangan. Sedangkan untuk kebutuhan *software* membutuhkan sistem operasi Windows 10 dan Kubuntu, Figma, Corel Draw X7, Android Studio, Android *Software Development Kit* (SDK), Flutter *Software Development Kit* (SDK), Visual Studio Code, Postman, Mozilla Firefox, MySQL, Apache, *shared hosting* dan sebuah domain. Aplikasi ini nantinya akan memiliki beberapa fitur, seperti menu hukum *tajwid* yang berisi materi dan contoh dari hukum *tajwid*, seperti *mad*, *mim* mati, *tanwin* atau *nun* mati, dan *idgham*, menu *juz amma* yang berisi *surah* pada *juz 30 Al-Quran* dengan ayat, terjemahan, serta ilustrasi yang menggambarkan maksud dari *surah* tersebut, dan fitur pemantauan a mengaji anak pada aplikasi melalui *bot* yang disediakan oleh Telegram dengan menggunakan *unique ID* yang didapatkan di dalam aplikasi *juz amma*.

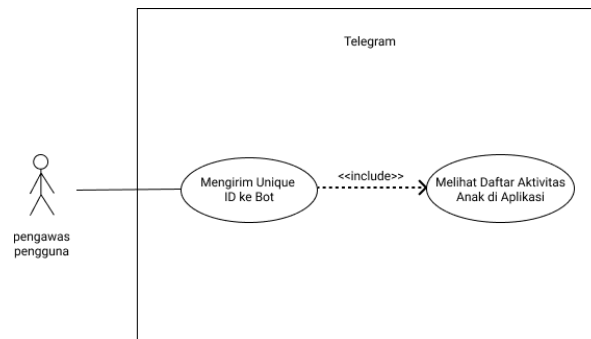
2.2 System Design

Kebutuhan dari tahap pertama dipelajari dalam tahap ini dan desain dari sistem disiapkan. Tujuan utama dari tahap desain adalah memberikan bentuk struktural yang sesuai dengan kebutuhan bahasa pemrograman. Dalam tahap ini visibilitas perangkat lunak dan cara kerja perangkat lunak dirancang (Malik, 2017). Tahapan ini dilakukan untuk membuat gambaran dari rancangan sistem yang akan dikembangkan. Hasil dari tahapan ini berupa *Use Case Diagram*, *Entity Relationship Diagram*, Rancangan *User Interface*.

2.2.1 Use Case Diagram



Gambar 1. *Use Case Diagram* Sistem Aplikasi Juz Amma



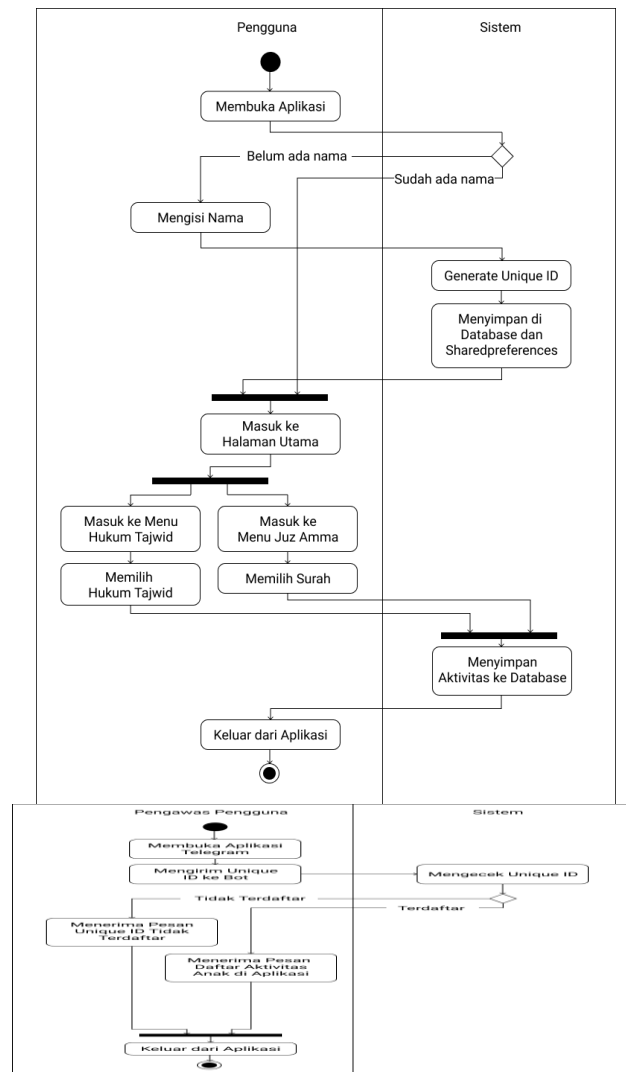
Gambar 2. *Use Case Diagram* Telegram

Setiap *use case* merepresentasikan kegiatan dari aktor, dimulai dengan tujuan aktor dan berakhir sampai tujuan itu terpenuhi. Terdapat 2 *Use Case* diagram pada aplikasi ini, yaitu untuk pengguna saat di aplikasi *juz amma* dan pengawas pengguna saat memantau aktivitas pengguna melalui Telegram.

Mengidentifikasi aktor adalah langkah awal dalam analisis *use case*. Setiap entitas eksternal yang berhubungan dengan sistem diwakili oleh sebuah aktor. Aktor bisa saja

seorang pengguna atau manusia, perangkat keras eksternal, atau subjek lain. Aktor pada sistem aplikasi *juz amma* hanya terdapat 1 aktor, yaitu pengguna. Pengguna dapat melakukan beberapa *use case* atau tindakan, seperti mengisi nama pengguna, mendapatkan *unique ID*, memilih menu belajar *tajwid*, memilih hukum *tajwid*, memilih menu membaca *juz amma*, dan memilih *surah* yang akan dibaca. Sedangkan untuk Aktor pada aplikasi Telegram hanya terdapat 1 aktor , yaitu pengawas pengguna. Pengawas pengguna dapat melakukan beberapa *use case* atau tindakan, seperti mengirim *unique ID* ke *bot* dan melihat daftar aktifitas anak di aplikasi.

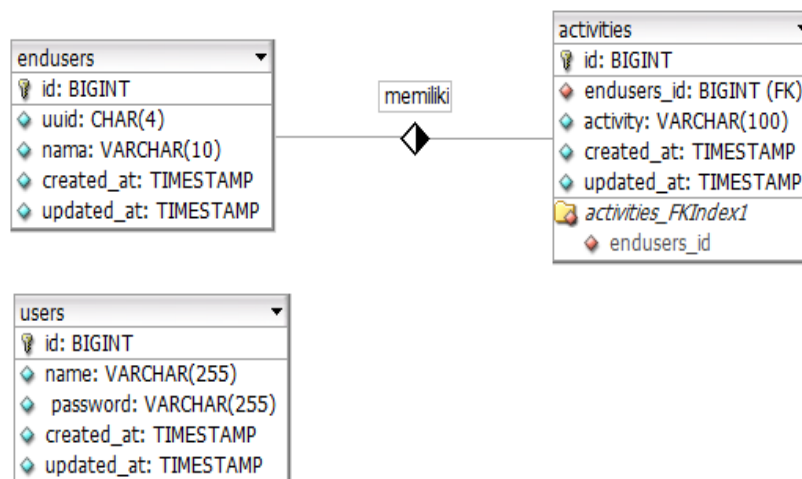
2.2.2 Activity Diagram



Gambar 3. Activity Diagram

Activity diagram sering digunakan untuk menguraikan dan memvisualisasikan *use cases* (Ahmad et al., 2019). Sebuah *Activity* diagram biasanya berisi *action* yang dilambangkan dengan persegi panjang, *initial node* berupa lingkaran hitam yang melambangkan dimulainya rangkaian *action*, *object flow* berupa tanda panah yang melambangkan arah dari satu *action* ke *action* lain, *decision node* berupa belah ketupat yang melambangkan pembagian 2 kondisi *action*, *fork node* yang berfungsi untuk membagi 1 *object flow* menjadi 2, *join node* yang berfungsi menggabungkan 2 *object flow* menjadi 1, dan *activity final node* berupa sebuah lingkaran hitam dengan celah putih yang melambangkan berakhirnya serangkaian *action*. *Activity* diagram pada aplikasi ini terdapat 2 *Activity* diagram, yaitu *Activity* diagram pengguna dan *Activity* diagram pengawas pengguna. *Activity* diagram pengguna berisi aktivitas yang dilakukan oleh pengguna di dalam aplikasi *juz amma*, sedangkan *Activity* diagram pengawas pengguna berisi aktivitas yang dilakukan oleh pengawas pengguna di aplikasi Telegram.

2.2.3 Entity Relationship Diagram



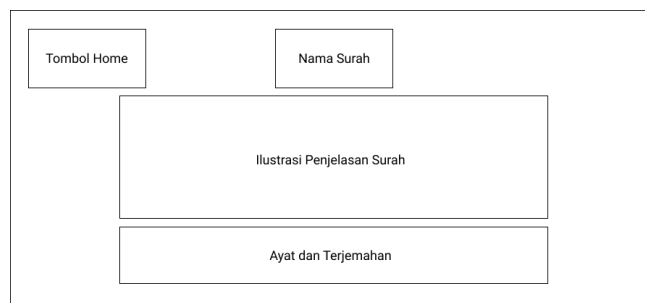
Gambar 4. Entity Relationship Diagram

Entity Relationship Diagram menggambarkan hubungan antar tabel atau entitas pada sebuah *database*, setiap hubungan atau relasi dari tiap tabel divisualisasikan dengan sebuah garis dan simbol. Diagram ini memiliki beberapa bagian, seperti entitas, atribut entitas, *primary key*, *foreign key*, dan relasi.

Entitas adalah hal atau konsep yang ada di dalam sebuah sistem, seperti peran misalnya *users*, objek misalnya profil, dan peristiwa misalnya *activities*. Entitas atau tabel pada *database* aplikasi ini terdapat 3 entitas, yaitu entitas *endusers*, *activities*, dan *users*. Entitas *endusers* berfungsi sebagai entitas yang berisi informasi dari pengguna, entitas ini terdapat beberapa atribut, yaitu id yang berperan sebagai *primary key*, *unique_id*, nama, *created_at*, dan *updated_at*. Sedangkan pada entitas *activities* yang berfungsi sebagai entitas yang mencatat seluruh aktivitas pengguna di aplikasi memiliki beberapa atribut, yaitu id yang berperan sebagai *primary key*, *enduser_id* yang berperan sebagai *foreign key* dari entitas *endusers*, *detail_activity*, *created_at*, dan *updated_at*. Entitas *endusers* dan entitas *activities* memiliki relasi *one-to-many*, yang dimana setiap 1 *user* dapat memiliki banyak *activity*. Entitas terakhir yaitu *users*, entitas ini hanya berfungsi untuk mengatur *user* admin, entitas ini terdapat beberapa atribut, yaitu id yang berperan sebagai *primary key*, *name*, *password*, *created_at*, dan *updated_at*.

2.2.4 Rancangan *User Interface*

User Interface (UI) adalah komponen penting dari sistem perangkat lunak apa pun, dengan kegunaan menjadi faktor kunci yang menentukan keefektifannya (Islam et al., 2020). *User Interface* berperan sebagai penghubung pengguna dengan aplikasi, selain harus memiliki *interface* yang menarik, tentu saja *developer* diharuskan membuat *interface* yang mudah digunakan, sehingga tidak hanya menarik untuk dilihat, namun harus juga bisa menyampaikan tujuan dari *interface* tersebut dan tidak membuat pengguna kesusahan saat menggunakannya. Berikut adalah rancangan *user interface* utama dari aplikasi pembelajaran *juz amma*.



Gambar 5. Tampilan *surah*

Tampilan *surah* pada gambar 5 adalah tampilan *surah*, tampilan ini berisi *surah Al-Fatihah* dan *surah-surah* pada *juz amma*, mulai dari *An-Nas* sampai *An-Naba'*. Setelah memilih salah satu dari *surah*, pengguna akan diarahkan menuju detail dari *surah* yang telah dipilih. Tampilan isi *surah* berisi detail dari *surah* yang telah dipilih, seperti nama *surah*, *ayat*, terjemahan, dan ilustrasi gambaran arti dari *surah*.

2.3 Development / Coding

Setelah sistem telah dirancang, tahapan selanjutnya adalah tahap *Development* (pengembangan) atau *Coding* (pengkodean). Tahapan ini adalah tahap terpanjang dari proses siklus pengembangan perangkat lunak, pada tahap ini dimulailah membangun seluruh sistem dengan menulis *code* dari bahasa pemrograman yang sudah dipilih, untuk bagian yang menghubungkan dengan database (*back-end*) menggunakan *framework* dari bahasa pemrograman PHP, yaitu Laravel. Sedangkan untuk bagian tampilan aplikasi mobile Android menggunakan *framework* dari bahasa pemrograman Dart, yaitu Flutter. Data-data *Al-Quran* didapatkan dari API (*Application Programming Interface*) yang terhubung dengan data *Al-Quran* yang disediakan oleh Kementerian Agama Indonesia pada *website quran.kemenag.go.id*. *Bot* Telegram yang digunakan untuk memantau aktivitas pengguna dibuat menggunakan *BotFather* yang disediakan oleh Telegram dan Laravel.

2.4 Testing

Testing merupakan tahapan penting untuk menguji kelayakan dari aplikasi yang telah dibuat dan memastikan aplikasi yang dibuat sesuai dengan perencanaan awal dalam hal antarmuka dan fungsi (Khafid & Putri, 2020). Tahapan ini bertujuan untuk mencari kesalahan atau *bug* pada aplikasi dan memastikan bahwa semua fungsi dalam aplikasi dapat berfungsi sesuai dengan keinginan, serta memenuhi rancangan aplikasi. Pengujian fungsional dari aplikasi dilakukan menggunakan metode *Black Box Testing*. Sedangkan untuk menguji aplikasi secara langsung kepada pengguna menggunakan *System Usability Scale*, pengujian dilakukan dengan tujuan mengetahui bagaimana tanggapan pengguna terhadap aplikasi. *System Usability Scale* (SUS) adalah kuesioner standar yang paling banyak digunakan untuk penilaian persepsi kegunaan (Lewis, 2018).

2.5 Deployment

Tahapan *deployment* adalah tahapan terakhir dari pengembangan perangkat lunak. Setelah melalui semua tahapan sebelumnya, aplikasi akan diunggah ke Google *Play Store* agar bisa digunakan oleh orang lain. Aplikasi ini nantinya akan selalu dipantau untuk mengetahui *review* dari pengguna, apakah ada kritik ataupun saran untuk pengembangan aplikasi pembelajaran *juz amma* ini.

3. HASIL DAN PEMBAHASAN

Berdasarkan pengembangan atau pengkodean aplikasi pembelajaran *juz amma* berbasis Android didapatkan hasil sebagai berikut :

3.1 User Interface Utama Aplikasi



Gambar 6. Tampilan Detail *Surah*

Gambar 6 merupakan implementasi dari rancangan *user interface surah*. Tampilan ini terdapat isi dari *surah* yang telah dipilih sebelumnya pada menu pilihan *surah*, terdapat beberapa isi *surah* seperti ilustrasi dari *surah*, *ayat surah* , *ayat surah* dalam huruf latin, terjemahan *surah*, dan *audio* dari tiap *ayat surah*. *Ayat* dari tiap *surah* ditampilkan satu persatu, tidak langsung semua *ayat* ditampilkan.

3.2 Testing

3.2.1 Pengujian *Black Box*

Pendekatan pengujian ini berfokus pada *input* yang masuk ke dalam aplikasi dan *output* yang dihasilkan. Pengujian tidak mencakup detail dalam seperti kode, logika server, dan metode pengembangan. Pengujian didasarkan pada *case* dan memeriksa aplikasi untuk

memvalidasi terhadap *case* yang telah ditentukan. Berikut adalah hasil dari pengujian *Black Box* yang ditampilkan pada Tabel 1.

Tabel 1. Hasil Pengujian Black Box

Test Case	Deskripsi Test Case	Hasil yang Diharapkan	Hasil Pengujian	Status
Memasukkan nama pengguna	Memastikan nama pengguna tersimpan di <i>database</i> dan muncul respon sukses	Nama pengguna tersimpan di <i>database</i> dan muncul respon sukses dengan keterangan, “Data Berhasil Disimpan”	Sesuai yang diharapkan	Sukses
Memilih salah satu menu hukum <i>tajwid</i>	Memastikan data aktivitas membuka salah satu menu hukum <i>tajwid</i> tersimpan di <i>database</i>	Aktivitas pengguna tersimpan di <i>database</i>	Sesuai yang diharapkan	Sukses
Memilih salah satu <i>surah</i>	Memastikan data aktivitas membuka salah satu <i>surah</i> tersimpan di <i>database</i>	Aktivitas pengguna tersimpan di <i>database</i>	Sesuai yang diharapkan	Sukses
Mengirim pesan “/activity <i>UniqueID</i> ” ke <i>Bot</i> Telegram JuzAmmaBot	Memastikan pesan yang dikirim <i>Bot</i> sesuai dengan <i>Unique ID</i> yang dikirim pengguna	Pesan yang dikirim <i>Bot</i> sesuai dengan <i>Unique ID</i> yang dikirim pengguna	Sesuai yang diharapkan	Sukses

Pengujian *Black Box* membantu menemukan kesalahan dalam fungsionalitas, kegunaan, dan fitur lainnya. Bentuk pengujian ini memberikan gambaran kinerja aplikasi

dan *output*-nya. Pengujian ini juga mengurangi risiko kegagalan aplikasi di pihak pengguna.

3.2.2 Pengujian *System Usability Scale*

Pengujian *System Usability Scale (SUS)* dilakukan untuk mengetahui respon dari pengguna terhadap sistem yang dibuat. Pengujian SUS berisi 10 pertanyaan dimana responden akan memberi peringkat setiap pertanyaan dari 1 hingga 5, berdasarkan seberapa setuju mereka dengan pernyataan yang mereka baca. 5 berarti sangat setuju, 4 berarti setuju, 3 berarti ragu-ragu, 2 berarti tidak setuju, dan 1 berarti sangat tidak setuju. Hasil pengujian SUS dapat dilihat pada Tabel 2.

Tabel 2. Hasil Pengujian *System Usability Scale*

No	Nama	Pertanyaan										X	Y	Total	Total * 2,5
		1	2	3	4	5	6	7	8	9	10				
1	Responden 1	1	3	3	2	5	3	3	3	3	4	10	10	20	50
2	Responden 2	4	2	4	2	4	2	4	2	2	4	13	13	26	65
3	Responden 3	4	1	5	3	5	1	5	1	2	3	16	16	32	80
4	Responden 4	4	3	4	4	4	2	5	3	4	4	16	9	25	62,5
5	Responden 5	4	3	3	2	4	2	4	5	4	5	14	8	22	55
6	Responden 6	4	2	4	3	4	2	4	2	4	2	15	14	29	72,5
7	Responden 7	4	2	4	1	5	2	4	2	4	3	16	15	31	77,5
8	Responden 8	4	2	5	2	5	1	5	1	5	3	19	16	35	87,5
9	Responden 9	4	1	5	2	4	2	4	1	4	2	16	17	33	82,5
10	Responden 10	5	1	5	2	5	1	4	1	5	1	19	19	38	95
11	Responden 11	4	3	4	4	4	2	4	3	3	4	14	9	23	57,5
12	Responden 12	4	3	4	4	3	3	3	2	3	5	12	8	20	50
13	Responden 13	4	1	5	1	5	1	4	1	5	1	18	20	38	95
14	Responden 14	5	2	4	2	4	2	4	2	4	2	16	15	31	77,5
15	Responden 15	4	2	4	2	4	2	3	3	4	5	14	11	25	62,5
16	Responden 16	4	3	4	2	4	3	4	2	4	4	15	11	26	65
17	Responden 17	5	2	4	2	5	2	5	1	3	3	17	15	32	80
18	Responden 18	1	2	5	2	4	1	4	1	4	1	13	18	31	77,5
19	Responden 19	5	2	4	2	5	2	4	2	4	3	17	14	31	77,5
20	Responden 20	4	1	5	1	4	2	5	1	4	2	17	18	35	87,5
Rata-rata															72,875

Hasil pengujian SUS mendapatkan nilai rata-rata dari sebesar 72,875. Produk yang mendapat nilai di atas 70 berarti dapat diterima (Bangor et al., 2009).

3.3 Deployment Aplikasi di Google Play Store

Aplikasi yang sudah selesai pengujian akan dipublikasikan di Google *Play Store*. Proses publikasi aplikasi di Google *Play Store* dilakukan melalui akun Google *Play Developer*. Sebelum aplikasi dipublikasikan akan dilakukan *review* terlebih dahulu oleh Google.

4. PENUTUP

Pengembangan aplikasi ini berhasil menghasilkan sebuah aplikasi pembelajaran *Juz Amma* berbasis Android. Berdasarkan 2 pengujian yang telah dilakukan terhadap aplikasi, bahwa dalam pengujian *Black Box* menunjukkan fungsi-fungsi dari aplikasi yang dibuat dapat berjalan dengan baik, sedangkan pada pengujian SUS didapatkan nilai rata-rata SUS sebesar 72,875 yang menunjukkan bahwa aplikasi sudah dalam kategori dapat diterima. Disimpulkan bahwa aplikasi pembelajaran *Juz Amma* berbasis Android ini dapat berjalan dengan baik sesuai dengan yang dirancang dan pengguna dapat menerimanya dengan mudah.

DAFTAR PUSTAKA

- Agarwal, P., Singhal, A., & Garg, A. (2017). SDLC Model Selection Tool and Risk Incorporation. *International Journal of Computer Applications*, 172(10), 6–10. <https://doi.org/10.5120/ijca2017915143>
- Ahmad, T., Iqbal, J., Ashraf, A., Truscan, D., & Porres, I. (2019). Model-based testing using UML activity diagrams: A systematic mapping study. *Computer Science Review*, 33(July), 98–112. <https://doi.org/10.1016/j.cosrev.2019.07.001>
- Andoyo, A., Aji, D. W., & Mukodimah, S. (2019). Sistem Aplikasi Pembelajaran Juz Amma Menggunakan Web Mobile untuk Meningkatkan Meningkatkan Minat Hafalan Surat Pendek Siswa MI Miftahul Huda Sendang Agung. *Jurnal Pendidikan Guru Madrasah Ibtidaiyah Al-Multazam*, 5(1), 38–46. <https://ojs.stitmaltazam.ac.id/index.php/JPGMI/article/view/60>
- Bangor, A., Staff, T., Kortum, P., Miller, J., & Staff, T. (2009). Determining what individual SUS scores mean: adding an adjective rating scale. *Journal of usability studies*, 4(3), 114–123. <https://uxpajournal.org/determining-what-individual-sus-scores-mean-adding-an-adjective-rating-scale/>

- Firdaus. (2016). Aplikasi Pembelajaran Juz Amma Berbasis Android. *Jurnal Teknik Informatika Aceh*, 1(2), 89–94. <https://garuda.ristekbrin.go.id/documents/detail/969636>
- Haris, M., Jadoon, B., Yousaf, M., & Khan, F. H. (2018). Evolution of Android Operating System: a Review. *Asia Pacific Journal of Contemporary Education and Communication Technology*, 4(1), 178–188. <https://doi.org/10.25275/apjcectv4i1ict2>
- Iqbal, S. Z., & Idrees. (2017). Z-SDLC Model A New Model For Software Development Life Cycle (SDLC). *International Journal of Engineering and Advanced Research Technology (IJEART)*, 3(2), 8. https://www.researchgate.net/publication/316317334_Z-SDLC_Model_A_New_Model_For_Software_Development_Life_Cycle_SDL
- Islam, M. N., Bouwman, H., & Islam, A. K. M. N. (2020). Evaluating web and mobile user interfaces with semiotics: An empirical study. *IEEE Access*, 8, 84396–84414. <https://doi.org/10.1109/ACCESS.2020.2991840>
- Khafid, B., & Putri, D. A. P. (2020). Pesma Apps as Android-based Integrated Applications for Mahasantri Pesma KH Mas Mansur UMS. *Khazanah Informatika: Jurnal Ilmu Komputer dan Informatika*, 6(2), 95–102. <https://doi.org/10.23917/khif.v6i2.10494>
- Lewis, J. R. (2018). The System Usability Scale: Past, Present, and Future. *International Journal of Human-Computer Interaction*, 34(7), 577–590. <https://doi.org/10.1080/10447318.2018.1455307>
- Malik, S. (2017). Software testing: Essential phase of SDLC and a comparative study of software testing techniques. *International Journal of System & Software Engineering*, 5(2), 38–45. https://www.academia.edu/40510082/Software_Testing_Essential_Phase_of_SDL_C_and_a_Comparative_Study_of_Software_Testing_Techniques
- Navita. (2017). A Study on Software Development Life Cycle & its Model. *International Journal of Engineering Research in Computer Science and Engineering*, 4(9), 1–4. https://www.technoarete.org/common_abstract/pdf/IJERCSE/v4/i9/Ext_96780.pdf
- Rahayu, S. L., & Fujiati, F. (2018). Penerapan Game Design Document dalam Perancangan Game Edukasi yang Interaktif untuk Menarik Minat Siswa dalam Belajar Bahasa Inggris. *Jurnal Teknologi Informasi dan Ilmu Komputer*, 5(3), 341. <https://doi.org/10.25126/jtiik.201853694>
- Solihin, R. (2020). Aplikasi Interaktif Tahfidz Al-Quran Juz Amma. *Jurnal Kajian Kritis Pendidikan Islam dan Manajemen Pendidikan Dasar*, 3(2), 1–11. https://doi.org/10.52484/as_sibyan.v3i2.182
- Sudarmilah, E., & Negara, M. G. (2013). Augmented Reality Edugame Senjata Indonesia.

Khazanah Informatika: Jurnal Ilmu Komputer dan Informatika, 1(1), 12–15.
<https://doi.org/10.23917/khif.v1i1.1176>

Tashildar, A., Shah, N., Gala, R., Giri, T., & Chavhan, P. (2020). Application Development Using Flutter. *International Research Journal of Modernization in Engineering Technology and Science*, 02(08), 1262–1266.
[https://irjmets.com/rootaccess/forms/uploads/APPLICATION DEVELOPMENT USING FLUTTER .pdf](https://irjmets.com/rootaccess/forms/uploads/APPLICATION_DEVELOPMENT_USING_FLUTTER.pdf)

Utomo, D., Sholeh, M., & Avorizano, A. (2017). Membangun Sistem Mobile Monitoring Keamanan Web Aplikasi Menggunakan Suricata dan Bot Telegram Channel. *Seminar Nasional Teknoka*, 2(2502), I 82-I 87.
<https://journal.uhamka.ac.id/index.php/teknoka/article/view/763>